

REMARKS

Applicant respectfully requests reconsideration of this application. Claims 1-14, 16-19, and 21-30 are pending. Claims 9-11 and 28-30 have been amended. Claims 31 and 32 have been added. No claims have been cancelled.

Therefore, claims 1-14, 16-19, and 21-32 are now presented for examination.

Claim Rejection under 35 U.S.C. §103

Aoshima, et al. in view of Young

The Examiner rejected claims 1, 2, 4-8, 12, 19, and 21-27 under 35 U.S.C. 103(a) as being unpatentable over U.S Patent No. 5,210,859 of Aoshima, et al. ("*Aoshima*") in view of U.S Patent No. 6,560,606 of Young ("*Young*").

It is again respectfully submitted that Aoshima and Young do not teach or suggest the elements of the claims. It is again submitted that there is no motivation shown for combining *Aoshima* and *Young*.

The arguments previously presented in the prior response remain relevant and the Applicant reiterates such arguments and requests reconsideration of these arguments. The Applicant further provides the following additional explanations in response to the rejections received in the current Office Action.

Specifically, it is submitted that, even if for the sake of argument it is assumed that the other claim elements are taught or suggested by *Aoshima*, it is respectfully submitted that the reference does not provide for modifying a module function in accordance with an inter-module dependency tree. Claim 1 reads as follows:

1. A method, comprising:
receiving requirements for a plurality of modules;

determining an inter-module dependency tree, the inter-module dependency tree being based on the requirements; and modifying a module function in accordance with the inter-module dependency tree.

As has been stated previously, Claim 1 thus provides for “receiving requirements” for modules, determining an inter-module dependency tree “based on the requirements”, and “modifying a module function in accordance with the inter-module dependency tree”.

With regard to the modification of a module function in accordance with an inter-module dependency tree, the Examiner has provided additional arguments based on certain portions of the reference. The Examiner refers to “In the fourth preferred embodiment, a description will now be made to such a case that reports on an execution history which have been stored in each branch of a module relation diagram may be referenced so as to be displayed in an execution order or an order opposite to the execution order (reverse execution order).” (*Aoshima*, col. 13, lines 32-38) This portion of *Aoshima* indicates that an execution history may be stored in each branch of a module relation diagram, but this does not indicate anything regarding any modification of a module. The execution history may be useful in the debugging operations described in *Aoshima*, but there is no teaching or suggestion regarding the modification of a module function in accordance with the inter-module dependency tree. Similarly, the office action cites to column 7, lines 62-65, which indicates that “In process 500, a user program (function) to be debugged is activated. In process 501, a calling relationship of a function corresponding to the relation diagram 73 shown in FIG. 7 is extracted.” This portion of *Aoshima* relates to the extraction of the calling relationship of a function, but

does not provide for modifying a module function in accordance with the inter-module dependency tree.

The Examiner further cites to the following portion description of Figure 16 of the reference:

FIG. 16 represents both tree data indicative of a module relation diagram and a relationship of the execution report in a case when the functions corresponding to the branches in the module relation diagram are executed a plurality of times. FIG. 16A represents an example of the module relation diagram in which a function "A" calls functions B and E, and the function B further calls functions C and D. It should be noted that the functions C and D are twice referred from the function B in the order of C, D, C and D. FIG. 16B represents a diagram for representing a relationship between the previously-described tree table and the executed report. In this case, although the functions C and D have been twice referred, one-line data has been recorded in the tree table due to the same relationship on the relation diagram.

Reference numerals 162 to 168 represent that the execution reports produced every time the respective functions are called are arranged in the calling order. The presence address of the respective reports are stored in the region 161 of the tree table when the respective reports are produced. In case of only one execution in the respective branches on the relation diagram, the address of the single report corresponds directly thereto. However, in such a case that the functions C and D are executed a plurality of times, a list is formed in which the addresses of the newly produced reports are successively arranged behind the addresses of the previously formed reports, and the head address of this list is stored into the region 161 as the address of the execution report.

(*Aoshima*, col. 12, line 40 to col. 13, line 2) In this portion of *Aoshima*, the reference illustrates data for a "module relation diagram", which provides some indication to a user

regarding how certain functions have been called. However, even assuming for sake of argument that this is a relevant provision, there is no teaching or suggestion of any modification of any action taken based on such information.

Therefore, Aoshima describes a structure that shows the existing calling relationships for modules. This is used for use in creating displays and reports that are used for debugging and program analysis. The module relations and related tree structure shown in Aoshima are clearly intended to be tools for error checking and debugging of programs. In Aoshima, the structure of a program already exists and the module relation diagram reflects this calling structure. As indicated in the quoted portion, information indicative of the module calling relationship is stored beforehand in a tree table, and according to this information the module relation diagram is displayed.

The examples provided by Aoshima show no modifications of any kind. In short, Aoshima provides a diagram as a record of calling relationships. As indicated in Aoshima, what is presented is a “program testing and debugging support system” (Aoshima, col. 1, lines 7-8) and specifically providing “for retrieving an execution history after the program is executed” (Aoshima, col. 1, lines 11-13)

As has been previously argued, it again is submitted that *Young* does not teach or suggest the elements of the claims missing from *Aoshima*, as argued above. *Young* specifically relates to computer processing of metered information regarding communication services. *Young* contains no teaching or suggestion of an inter-module dependency tree based on module requirements, or modifying a module function in accordance with an inter-module dependency tree.

For these and other reasons, *Aoshima* and *Young*, separately or in combination, do not teach or suggest the elements of the independent claims presented in the application. The rejected claims are dependent claims that are allowable as being dependent on the allowable base claims.

It is submitted that the above arguments also applies to independent claims 12, 19, and 24 and such claims are thus also allowable. The remaining claims are dependent claims and are allowable as being dependent on the allowable base claims.

Claim Rejection under 35 U.S.C. §103

Aoshima, et al. in view of Young and APA

The Examiner rejected claims 3, 9-11, 13, 14, 16-18, and 28-30 under 35 U.S.C. 103(a) as being unpatentable over Aoshima in view of Young as applied to claim 1 and further in view of subject matter alleged to be admitted prior art (APA).

It is submitted that the rejected claims are allowable as being dependent on the allowable base claims, the independent claims having been shown to be allowable in the above arguments.

The arguments presented previously are hereby resubmitted. It is again submitted that there is no motivation shown for combining Aoshima and Young with the alleged APA.

In addition, with regard to the specific operations, claims 9-11, as amended herein, are as follows:

9. The method of claim 1 wherein modifying a module function comprises initializing the module function in accordance with the inter-module dependency tree.

10. The method of claim 1 wherein modifying a module function comprises reconfiguring the module function in accordance with the inter-module dependency tree.
11. The method of claim 1 wherein modifying a module function comprises shutting down the module function in accordance with the inter-module dependency tree.

In addition to any other differences, it is submitted that the cited references do not contain any of these claim elements. With regard to these elements, the Office Action cites to the background of the present application: “Modules in a device may not be not totally independent – some modules may use services provided by other modules. Therefore, during initialization, reconfiguration and shutdown the modules have to be acted upon in a proper sequence, as implied by the aforementioned dependencies.” However, it is submitted that this misses the point as this is simply a statement of the problem, i.e., that during these stages modules need to be acted on in accordance with their dependencies. This statement of the problem does not teach or suggest a solution to the problem. In relation to claims 9-11, the modification of a module function in accordance with the inter-module dependency tree may include initializing, reconfiguring, or shutting down a module function in accordance with the inter-module dependency tree.

It is submitted that these arguments also apply to claims 28-30, as amended herein, and to new claims 31 and 32.

Conclusion

Applicant respectfully submits that the rejections have been overcome by the amendment and remark, and that the claims as amended are now in condition for

allowance. Accordingly, Applicant respectfully requests the rejections be withdrawn and the claims as amended be allowed.

Invitation for a Telephone Interview

The Examiner is requested to call the undersigned at (303) 740-1980 if there remains any issue with allowance of the case.

Request for an Extension of Time

The Applicant respectfully petitions for a one-month extension of time to respond to the outstanding Office Action pursuant to 37 C.F.R. § 1.136(a). A check for the required fee under 37 C.F.R. § 1.17 for such an extension is enclosed herewith.

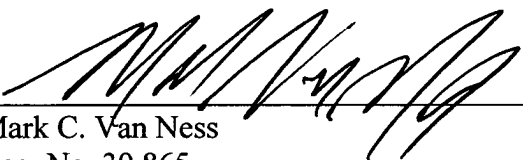
Charge our Deposit Account

Please charge any shortage to our Deposit Account No. 02-2666.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: 5/11/06



Mark C. Van Ness
Reg. No. 39,865

12400 Wilshire Boulevard
7th Floor
Los Angeles, California 90025-1026
(303) 740-1980